

一、单选题(共 25 题，共 50 分)

1. 下列有关分治算法思想的描述不正确的是？（ ）

- A. 将问题分解成的子问题具有相同的模式。
- B. 将问题分解出的各个子问题相互之间有公共子问题。
- C. 当问题足够小时，可以直接求解。
- D. 可以将子问题的求解结果合并成原问题的解。

2. 斐波那契数列前 n 项是 1,1,2,3,5,8,13.....

补全下面程序代码，求第 n 项斐波那契数列的值。

```
def fib(n):  
    if ____:  
        return 1  
    else:  
        return fib(n-1) + fib(n-2)
```

- A. n == 2
- B. n < 2
- C. n == 3
- D. n < 3

3. 用分治法求两个数的最大公约数，代码和运行结果如下，请选择合适的代码完

成程序？（ ）

```
def fun(m, n):  
    if m % n == 0:  
        return n  
    else:  
        return _____  
print(fun(98, 56))
```

程序运行结果如下：

14

- A. fun(n, m-n)
- B. fun(n, m%n)
- C. fun(m, m-n)
- D. fun(m, m%n)

4. 列出第三方库的详细信息 pip 指令是？（ ）

- A. pip install <第三方库名>
- B. pip download<第三方库名>
- C. pip show<第三方库名>
- D. pip list<第三方库名>

5. 想要在 Python 中导入 math 模块中的 sqrt(x)函数，下列程序段错误的是？（

）

- A. import math
math.sqrt(x)
- B. from math import sqrt as t
t(x)
- C. import math as m

m.sqrt(x)
import math
D. sqrt(x)

6. 关于求解“找出所有满足各位数字之和等于 8 的三位数”时，在下列数值范围内，算法执行效率最高的是？（ ）

- A. 0—999
- B. 0—1000
- C. 100—800
- D. 107—800

7. 已知，从 1 到 n 的连续自然数相乘的积叫做阶乘，用符号 $n!$ 表示，比如 $3! = 1 \times 2 \times 3$ ，规定 $0! = 1$ 。那么用递归算法求 n 的阶乘，递归式正确的是？（ ）

- A. $f(0)=1, n=n*(n-1)$
- B. $f(0)=1, f(n)=f(0)*f(n-1)$
- C. $f(0)=1, f(n)=f(n)*f(n-1)$
- D. $f(0)=1, f(n)=n*f(n-1)$

8. 当 n 为 6 时，运行下列 Python 程序后的结果是？（ ）

```
def f(n):  
    if n<=2:  
        return 1  
    else:  
        return f(n-1)+f(n-2)  
  
n=int(input("请输入一个正整数："))  
print(f(n))
```

- A. 5
- B. 8
- C. 11
- D. 13

9. 下列程序中，当 $n=4$ 时，返回结果为？（ ）

```
def x(n):  
    if n==1:  
        return 1  
    else:  
        return n+x(n-1)
```

- A. 12
- B. 11
- C. 10
- D. 9

10. 下列选项中，不属于递归特性的是？（ ）

- A. 递归函数一定包含条件控制语句
- B. 递归算法体现了大事化小的思想
- C. 递归效率不高，递归层次过多会导致栈溢出
- D. 递归算法代码简洁，效率较高

11. 对于下列递归式子，当 $n=4$ 时， F 的值是？（ ）

$F(n)=F(n-1)*2$
 $F(1)=2$

- A. 2

- B. 8
- C. 16
- D. 32

12. 下面函数实现的功能是？（ ）

```
def mi(x, n):  
    if n == 0:  
        return 1  
    else:  
        return x*mi(x, n-1)
```

- A. 计算 x 的 n 次方
- B. 计算 n 的 x 次方
- C. 计算 x!*n
- D. 计算 x*n!

13. def afun(n):
s=1
for i in range(1,n+1):
s*=i
return s

对以上代码解释错误的是？（ ）

- A. 程序正常运行时，afun(n)函数的作用是求 n 的阶乘
- B. 程序正常运行时，afun(n)函数的作用是求 n+1 的阶乘
- C. s 是局部变量
- ☐ range()函数是 Python 内置函数
- D.

14. 关于下列代码解释错误的一项是？（ ）

```
def ZhongYao(a:str,b:int)->str:  
    word=a*b  
  
    print('重要的事情说{}遍：{}'.format(b,word))  
    return ZhongYao
```

ZhongYao('学习','3')

- A. 参数 a 的数据类型必须是字符串
- B. 参数 b 的数据类型必须是整型
- C. 函数指定了返回值的数据类型是字符串

程序正常运行，输出结果是：

- D. 重要的事情说 3 遍：学习学习学习

15. 下列代码的运行结果是？（ ）

```
def exchange(a,b):  
    a,b=b,a  
    print(a,b)  
a,b=3,5  
exchange(a,b)  
print(a,b)
```

- ☐ 5 3
- A. 5 3
- ☐ 3 5
- B. 3 5
- ☐ 5 3
- C. 3 5
- ☐ 5
- ☐ 3
- D. 3 5

16. 下列代码的运行结果是？（ ）

```
def car_show(cars):  
    for car in cars:  
        print('good',car)  
car_show(['BYD','Haval','Wuling'])
```

- ☐ good ['BYD','Haval','Wuling']
- A. 'good','BYD'
- ☐ 'good','Haval'
- B. 'good','Wuling'
- ☐ good BYD Haval Wuling
- C. good BYD
- ☐ good Haval
- D. good Wuling

17. 函数中定义了 4 个参数，其中 2 个参数都指定了默认值，见下面代码，那么在

调用函数时参数个数最少是？（ ）

```
def SiBianXing(a,b,c=5,d=8):  
    ZhouChang=a+b+c+d  
    return ZhouChang  
SiBianXing( ? )
```

- ☐ 0
- A. 2
- ☐ 2
- B. 1
- ☐ 1
- C. 3
- ☐ 3
- D.

18. 执行如下程序，以下选项中哪一项是错误的？（ ）

```
def f(a,b):  
    c=a+b*2  
    b=a  
    return c  
a=1  
b=2  
c=f(a,b)+b
```

- ☐ 该函数名称为 f
- A.

- ☐ 执行该程序后，变量 a 的值为 1
- B.
- ☐ 执行该程序后，变量 b 的值为 2
- C.
- ☐ 执行该程序后，变量 c 的值为 6
- D.

19. 以下哪种情况下的代码块适合设计成函数？（ ）

- ☐ 复杂的功能块
- A.
- ☐ 难以看懂的功能块
- B.
- ☐ 有性能要求的功能块
- C.
- ☐ 会多次用到的功能块
- D.

20. 使用位置实参的方式调用函数时，下列哪种情况下程序一定会出错？（ ）

- ☐ 传入的实参个数比形参个数多
- A.
- ☐ 部分形参赋以默认值
- B.
- ☐ 函数调用时，重新给赋予了默认值的形参传入新值
- C.
- ☐ 部分实参以关键字赋值的方式去调用函数
- D.

21. 下列函数定义中，正确的是？（ ）

- ☐ `def myfunc(a,b,c)`
`print(myfunc(a,b))`
- A.
- ☐ `def myfunc(a=1,b,c):`
`print(myfunc(a,b,c))`
- B.
- ☐ `def myfunc(a,b=1,c):`
`print(myfunc(a,b,c))`
- C.
- ☐ `def myfunc(a,b,c=1):`
`print(myfunc(a,b,c))`
- D.

22. 调用函数的时候，所使用的参数是？（ ）

- ☐ 实参
- A.
- ☐ 位置参数
- B.
- ☐ 关键字参数
- C.

- ☐ 不定长参数
- D.

23. 对于如下自定义的函数：

```
def myfunc(a,b,c):  
    return a+b+c
```

以下调用该函数的语句中，能正确执行的是？（ ）

- ☐ myfunc(1,2)
- A.
- ☐ myfunc(a=3,2,1)
- B.
- ☐ myfunc(3,b=2,1)
- C.
- ☐ myfunc(3,2,c=1)
- D.

试题编号：20230713-xm-005

24. 使用*args 和**kwargs 形参来定义函数时，正确的说法是？（ ）

- ☐ 使用*args 代表列表，使用**kwargs 代表元组
- A.
- ☐ 可以在调用函数时传入多个实参
- B.
- ☐ 使用*args 代表元组，使用**kwargs 代表列表
- C.
- ☐ 使用*args 代表字典，使用**kwargs 代表列表
- D.

25. 关于函数参数的默认值设置，描述不正确的是？（ ）

- ☐ 定义函数时，可以为某形参设定默认值
- A.
- ☐ 如果为某形参设定了默认值，则在调用函数时就不能再为该形参传入新
- B. 值
- ☐ 如果为某形参设定了默认值，在调用函数时还可以为该形参再传入新值
- C.
- ☐ 关键字形参就是一种为参数设置默认值的方式
- D.

二、判断题(共 10 题，共 20 分)

26. 分治算法是把一个大问题分解为若干个规模较小、性质相同的子问题。最后子问题可以简单地直接求解，将所有子问题的解合并起来就是原问题的解。

（ ）

☐ 正 ☐ 错

确 误

27. pip 联网直接下载安装第三方库时，只能临时更换下载源，不能更改默认的下
载源。（ ）

☐ 正 ☐ 错

确 误

28. 运行以下代码的时间复杂度为 $O(n^2)$ 。（ ）

```
k=0
n=11
for i in range(n):
    k=k+1
for j in range(n):
    k=k+2
print(k)
```

☐ 正 ☐ 错

确 误

29. 对于递归而言，递推与回归，二者缺一不可。（ ）

☐ 正 ☐ 错

确 误

30. 递归算法必须确保，需要解决的问题可以转化为一个或多个子问题来求解，这
些子问题的求解方法与原问题相同，只是在数量和规模上不同，而且每次递归
调用时，问题规模都能够缩小。（ ）

☐ 正 ☐ 错

确 误

31. 当函数运行结束后，局部变量的值依然存在，下次函数调用可以继续使用。（
）

☐ 正 ☐ 错

确 误

32. 内置函数不需要使用关键字 import 导入就可以使用，自定义函数可以先调用
后定义。（ ）

☐ 正 ☐ 错

确 误

33. 编程时，需要把实现复杂功能的代码封装起来，设计成函数。（ ）

☐ 正 ☐ 错

确 误

34. 将实现特定功能的代码块设计成函数，有助于提高整个程序的运行速度。

()

☐ 正 ☐ 错

确 误

35. 如果允许形参设置默认值，则函数定义时要先列出没有默认值的形参，再列出有默认值的形参。()

☐ 正 ☐ 错

确 误

三、编程题(共 3 题，共 30 分)

36. 统计高分问题

小张在学习对分查找后，想利用该思路来实践一下，加深对该算法的理解。

于是他按老师的要求，统计期末考试成绩，总分比他高的同学有多少个？

他的思路是，首先将全区所有学生的成绩降序排序，然后利用对分查找思想解决。他编写的代码如下，请完善划线处的代码。

`cj =[654,643,621,612,609,606,...]` #学生成绩存列表 cj 并已经降序排序

```
def dfsearch(key):  
    i =0; j=len(cj)-1  
    while i<=j:  
        m =(i+j)//2  
        if cj[m]>key:  
            _____①  
        else:  
            j-= 1  
    return _____②
```

`key = int(input('输入待查找的成绩'))`

`pos =_____③`

`print('高于',key,'分的人数有',pos)`

试题难度：一般

试题解析：

评分标准：

(1) `i += 1` 或等价表达式；（2 分）

(2) `i` 或等价表达式；（3 分）

(3) `dfsearch(key)`或等价表达式。（3 分）

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：0

是否评分：已评分

评价描述：

37. 比岁数问题

有 5 个人坐在一起，问第五个人多少岁？他说比第 4 个人大 2 岁。问第 4 个人岁数，他说比第 3 个人大 2 岁。问第三个人，又说比第 2 人大两岁。问第 2 个人，说比第一个人大两岁。

最后问第一个人， he 说是 10 岁。请问第五个人多大？

请运用递归算法完成以下代码。

```
def age(n):  
    if _____①:   
        c = 10  
    else:   
        c = _____②  
    return c  
print(age(_____③))
```

试题难度：一般

试题解析：

评分标准：

(1) $n == 1$ 或等效答案；（3 分）

(2) $\text{age}(n - 1) + 2$ 或等效答案；（4 分）

(3) 5 或等效答案。（3 分）

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：3

是否评分：已评分

评价描述：

38. 数学证明问题

上世纪中期，校园里流行一种数学游戏，其规则十分简单：任意取一个自然数，若它是奇数，则把它乘以 3 加上 1，若它是偶数，则除以 2。按此规则进行下去，经过若干步的计算就能得到自然数 1。这个数学现象至今都没有得到完全的证明，但是可以编程来验证它的正确性。

实现上述功能的部分程序代码如下，请补充完成空白处的内容。

```
step = 1  
midlist = []  
  
def subwork(n):  
    global step  
    if n == 1:
```

```
    return step
elif n % 2 == 0:
    n = _____①
    midlist.append(n)
    step += 1
    subwork(_____②)
    return step
elif n % 2 != 0:
    n = _____③
    midlist._____④
    step += 1
    subwork(_____⑤)
    return step
```

```
cur_num = int(input("请输入一个数字 : "))
subwork(cur_num)
print(midlist)
print(step)
```

试题难度：一般

试题解析：

评分标准：

- (1) $n / 2$ 或等效答案；（3分）
- (2) n 或等效答案；（2分）
- (3) $n * 3 + 1$ 或等效答案；（3分）
- (4) `append(n)`或等效答案；（2分）
- (5) n 或等效答案。（2分）

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：10

是否评分：已评分

评价描述：